doi：10.13413/j.cnki.jdxblxb.2023236

屋型拓扑粒子群优化算法与工程优化问题求解

高铭晗[1]，王丽敏[2]，黄锐露[2]，张宇飞[3]，李明洋[4]
（1. 长春工业大学 计算机科学与工程学院，长春 130012；2. 广东财经大学 信息学院，广州 510320；
3. 长春大学 计算机科学技术学院，长春 130022；4. 长春工业大学 经济管理学院，长春 130012）

摘要：针对粒子群优化算法在优化复杂工程问题时存在搜索效率低和易陷入局部最优的问题，提出一种屋型拓扑粒子群优化算法. 该算法通过提出屋型拓扑和设计适应其特性的位置更新策略，改善粒子群优化算法信息传递和交流方式，提升算法的收敛速率和全局优化能力. 在基准函数上的对比实验结果表明，屋型拓扑粒子群算法的寻优精度、收敛速度和稳定性均优于其他 4 种改进算法. 在 3 个实际工程优化问题上的仿真实验结果进一步验证了该算法的有效性和实用性.

关键词：屋型拓扑；粒子群优化算法；工程优化问题；基准函数；仿真实验
中图分类号：TP301.6 **文献标志码**：A **文章编号**：1671-5489(2024)06-1384-07

House Topology-Based Particle Swarm Optimization Algorithm and Solution to Engineering Optimization Problems

GAO Minghan[1], WANG Limin[2], HUANG Ruilu[2], ZHANG Yufei[3], LI Mingyang[4]
(1. *School of Computer Science & Engineering, Changchun University of Technology, Changchun* 130012, *China*;
2. *School of Information Science, Guangdong University of Finance and Economics, Guangzhou* 510320, *China*;
3. *School of Computer Science and Technology, Changchun University, Changchun* 130022, *China*;
4. *School of Economics and Management, Changchun University of Technology, Changchun* 130012, *China*)

Abstract: Aiming at the problems of low search efficiency and susceptibility to local optima in particle swarm optimization algorithm for optimizing complex engineering problems, we proposed a house topology-based particle swarm optimization algorithm. By proposing a house topology and designing a position update strategy tailored to its characteristics, the algorithm improved the information transmission and communication methods of particle swarm optimization algorithm, thereby enhancing the convergence rate and global optimization capability of the algorithm. The comparative experimental results on benchmark functions show that the optimization accuracy, convergence speed, and stability of the house topology-based particle swarm optimization algorithm are superior to the other 4 improved algorithms. The simulation results on 3 real-world engineering optimization problems further validate the effectiveness and practicality of the proposed algorithm.

Keywords: house topology; particle swarm optimization algorithm; engineering optimization problem; benchmark function; simulation experiment

收稿日期：2023-06-12.

第一作者简介：高铭晗(1998—)，男，汉族，硕士研究生，从事智能优化和联邦学习的研究，E-mail：minghangao@outlook.com.
通信作者简介：王丽敏(1975—)，女，汉族，博士，教授，从事人工智能和数据挖掘的研究，E-mail：20211016@gdufe.edu.cn.

基金项目：国家社会科学基金(批准号：22BTJ057；24BTJ041；24CTJ034)、教育部人文社科基金(批准号：24YJAZH042)和吉林省自然科学基金(批准号：20200201164JC).

现代工程领域优化问题通常包含多种变量和约束条件，具有复杂性和多维性，求解难度较大[1]. 目前已提出了多种优化方法，如梯度下降法和牛顿法等. 这些方法具有收敛速度快、可解释性强等优点，但有时面临初始条件敏感和易陷入局部最优等问题，从而很难找到可行解. 群体智能优化算法为这类问题的求解提供了解决方案.

粒子群优化(particle swarm optimization，PSO)算法是工程应用中最广泛的群体智能优算法之一[2]，其通过种群粒子间的相互协作和信息共享寻找问题最优解，因简单易行和全局搜索能力较强而备受关注. 但在处理复杂工程优化问题时，PSO 算法存在搜索效率低和搜索能力不足等问题. 因此，改进 PSO 算法的性能，进一步提升算法的全局搜索能力和收敛速度至关重要. 目前已提出了多种改进的 PSO 算法，主要包括改进拓扑结构[3-5]、动态参数调整[6-8]和混合算法[9-11]等. 其中，改进拓扑结构的方法通过改变粒子之间的信息交流方式，增强算法多样性. 动态参数调整的方法根据算法的进程或粒子的表现动态调整算法参数，平衡搜索能力. 混合算法的方法通过引入其他优化算法机制，利用各算法的优势弥补单一算法不足. 上述方法在一定程度上改善了 PSO 算法的局限性，提升了其全局搜索能力和收敛速度.

本文提出一种屋型拓扑粒子群优化(house topology-based particle swarm optimization，HTPSO)算法. 该算法通过设计屋型拓扑结构，优化了 PSO 算法种群组织方式，扩展了粒子知识来源. 设计适应屋型拓扑的位置更新策略，提升算法收敛速率和全局优化能力. 在仿真测试中，利用多个基准函数验证了 HTPSO 算法具有收敛速度快、精度高等优势. 在 3 个工程优化问题中，相较于对比算法，HTPSO 算法展现了更具优势的综合优化性能.

1　粒子群优化算法

PSO 算法在问题求解时，通常采用随机初始化方法构造初始粒子群，然后利用迭代位置更新方法，根据种群内部经验不断自我提升，使种群整体趋向最有可能存在全局最优解的方向，从而找到具体优化问题的近似最优解. 假设一个具有 D 维搜索空间的优化问题，当使用具有 N 个种群粒子的 PSO 算法进行求解时，其第 i 个粒子的位置向量可表示为 $\boldsymbol{X}_i=(x_{i,1},x_{i,2},\cdots,x_{i,D})$，速度向量可表示为 $\boldsymbol{V}_i=(v_{i,1},v_{i,2},\cdots,v_{i,D})$. 在一次迭代中，PSO 算法粒子的位置更新公式和速度更新公式分别为

$$\boldsymbol{V}_i(t+1)=w\boldsymbol{V}_i(t)+c_1r_1(\boldsymbol{X}_{i,\text{best}}(t)-\boldsymbol{X}_i(t))+c_2r_2(\boldsymbol{X}_{\text{gbest}}(t)-\boldsymbol{X}_i(t)),\tag{1}$$

$$\boldsymbol{X}_i(t+1)=\boldsymbol{X}_i(t)+\boldsymbol{V}_i(t+1),\tag{2}$$

其中：$\boldsymbol{X}_{i,\text{best}}$表示粒子 i 到当前迭代 t 为止经过的具有最佳适应度值的位置；$\boldsymbol{X}_{\text{gbest}}$表示整个种群的最佳位置记录；$c_1$ 和 c_2 分别表示认知和社会学习因子，用于调整粒子受到个体和全局最优吸引的强度，通常均取值为 2；r_1 和 r_2 为两个均匀分布在 0～1 内的随机数；w 为线性递减的惯性权重，可表示为

$$w=w_{\max}-\frac{t(w_{\max}-w_{\min})}{t_{\max}},\tag{3}$$

$w_{\max}$和 $w_{\min}$分别为最大和最小惯性权重值，$t_{\max}$为最大迭代次数.

2　屋型拓扑粒子群优化算法

2.1　屋型拓扑

PSO 算法中粒子受种群最优个体位置和自身历史最优位置的吸引，可能会导致种群多样性降低，从而快速收敛于局部最优解附近. 本文提出屋型拓扑结构组织粒子协作完成寻优任务.

屋型拓扑是一种多层的拓扑结构，其根据类似现实世界中的房屋形态展开运作. 在一个屋型拓扑种群中，粒子将被划分为屋顶、屋檐、墙壁三类. 具有更好适应度值的粒子，会被分配在更高层中. 位于屋顶的粒子作为领导者，能直接或间接管理和引导其下的诸多粒子. 即屋顶的粒子可以引导屋檐中的粒子，而其他层的粒子同样能引导其下一层的粒子. 通过多个层次之间的协作，实现对整个种群知识的传递，从而能比 PSO 算法拓扑更恰当地分配粒子任务. 图 1 为具有 15 个粒子的屋型拓扑种群示意图.

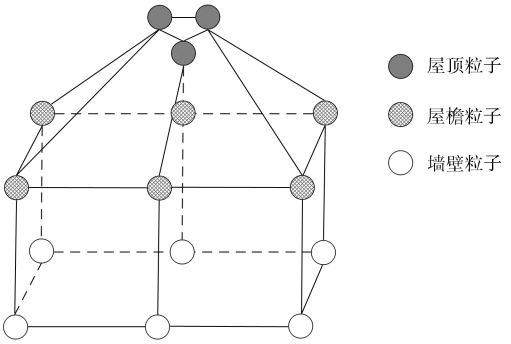


图1 屋型拓扑示意图

Fig. 1 Schematic diagram of house topology

屋型拓扑的PSO算法有以下属性：

1) 每个粒子被分配到一个特定的层；

2) 粒子适应度值越好，其所在的层次越高，全局最佳粒子位于屋顶，最差的粒子位于墙壁；

3) 同层粒子具有相近的适应度值.

2.2 改进位置更新策略

PSO算法种群整体使用相同的位置更新方法，可能导致不同粒子的潜力无法充分发挥，影响算法的优化性能. 本文设计3种位置更新策略，以适应屋型拓扑的多层运作结构，从而为不同层次的粒子分配合适的任务，以增强算法性能.

屋顶粒子具有最好的适应度值，其位置可能具有较强的全局表示性和优势性. 因此，屋顶粒子的位置更新方法应能加快收敛速度，并精细开发种群已知的最优区域. 屋顶粒子的速度和位置更新公式分别为

$$\boldsymbol{V}_i(t+1)=r_1\sin(2\pi(1-\mu))(\boldsymbol{X}_i(t)-\boldsymbol{X}_{\mathrm{gbest}}(t)), \tag{4}$$

$$\boldsymbol{X}_i(t+1)=\boldsymbol{X}_{\mathrm{gbest}}(t)+\boldsymbol{V}_i(t+1), \tag{5}$$

其中：r_1 和 μ 为两个均匀分布在0～1内的随机数；$\boldsymbol{X}_{\mathrm{gbest}}$ 为当前时刻内种群历史最优经验位置. 通过引入正弦扰动，扩展了屋顶粒子速度更新的行为方式，为其位置更新提供了更泛化性的参考，一定程度上避免了在自身最优区域陷入迟滞的可能.

屋檐粒子具有次好的适应度值，在寻优过程中起到承上启下、串联全局的作用. 因此，屋檐粒子的位置更新方法应根据屋顶粒子的指导，充分吸收屋顶粒子的历史优秀经验，从而在保有适当多样性的同时，辅助屋顶粒子实现对解空间的进一步开发. 屋檐粒子速度和位置更新公式分别为

$$\boldsymbol{V}_i(t+1)=w\boldsymbol{V}_i(t)+c_1r_2(\boldsymbol{X}_{i,\mathrm{best}}(t)-\boldsymbol{X}_i(t))+c_2r_3(\boldsymbol{P}_i(t)-\boldsymbol{X}_i(t)), \tag{6}$$

$$\boldsymbol{X}_i(t+1)=\boldsymbol{P}_i(t)+\boldsymbol{V}_i(t+1), \tag{7}$$

其中：$\boldsymbol{X}_{i,\mathrm{best}}$ 为粒子 i 的历史最优位置；r_2 和 r_3 为两个均匀分布在0～1内的随机数；w 为惯性权重，取值如式(3)所示；$\boldsymbol{P}_i$ 为屋顶共识位置，可表示为

$$\boldsymbol{P}_i(t)=\frac{\boldsymbol{X}_{\mathrm{roof1}}(t)+\boldsymbol{X}_{\mathrm{roof2}}(t)+\boldsymbol{X}_{\mathrm{roof3}}(t)}{3}, \tag{8}$$

$\boldsymbol{X}_{\mathrm{roof1}}$，$\boldsymbol{X}_{\mathrm{roof2}}$，$\boldsymbol{X}_{\mathrm{roof3}}$ 分别表示3个随机选择的不同屋顶粒子的历史最优位置. 通过引入屋顶共识位置，屋檐粒子速度更新的一个分量将在有效继承屋顶粒子经验的同时，避免过快收敛到某一局部最优位置附近.

墙壁粒子的适应度值较差，可能蕴含的全局最优信息较少. 因此，本文采用重组位置更新方法，帮助墙壁粒子快速脱离当前区域，并有效增强种群多样性. 墙壁粒子的位置更新公式为

$$\boldsymbol{X}_i(t+1)=\boldsymbol{E}_i(t)+r_4\eta(\boldsymbol{X}_{e,\mathrm{best}}(t)-\boldsymbol{X}_i(t)), \tag{9}$$

$$\boldsymbol{E}_i(t)=\frac{\boldsymbol{X}_{e,\mathrm{best}}(t)+\boldsymbol{X}_i(t)}{2}, \tag{10}$$

其中：$\boldsymbol{X}_{e,\mathrm{best}}$ 表示随机选择的一个屋檐粒子的历史最优位置；$\eta=0.6$；r_4 为均匀分布在0～1内的随机

数；$\boldsymbol{E}_i$ 为重组位置，由式(10)获得.

2.3　HTPSO 算法实现过程

HTPSO 算法的实现流程如图 2 所示.

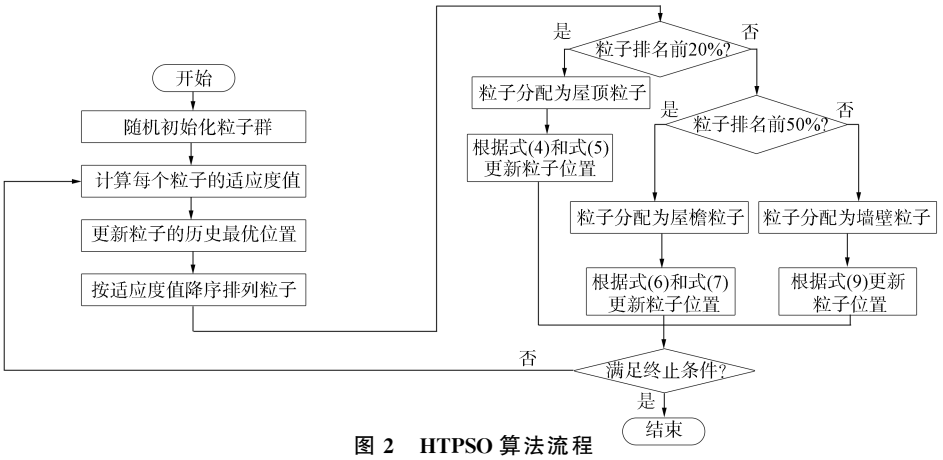


图 2　HTPSO 算法流程

Fig. 2　Flow chart of HTPSO algorithm

3　仿真实验

为验证本文方法的有效性，下面在多个基准函数和 3 个工程优化问题上进行仿真实验. 实验设备为搭载 Intel(R) Xeon(R) CPU E5-2666 v3 @ 2.90 GHz 的工作站，内存为 32 GB，操作系统为 Microsoft Windows10 64 位，仿真软件为 MATLAB R2022a.

3.1　基准函数测试

本文选择 CEC2017 技术报告[12]中的 6 个基准函数进行测试，各基准函数信息列于表 1.

表 1　各基准函数信息

Table 1　Information of each benchmark function

函数	函数名称	最优值	函数	函数名称	最优值
F_1	位移和旋转的 Bent Cigar 函数	100	F_4	混合函数 3(N=3)	1 300
F_2	位移和旋转的 Zakharov 函数	300	F_5	混合函数 4(N=4)	1 400
F_3	混合函数 2(N=2)	1 200	F_6	混合函数 6(N=5)	1 800

在基准函数上，将 HTPSO 算法与 4 种先进的改进算法进行性能比较，包括两个同类改进算法：自适应加权粒子群优化器(AWPSO)[13]和修正的粒子群优化(MPSO)算法[14]. 两个其他优化算法的改进算法：具有性能增强策略的鸡群优化(PECSO)算法[15]和正弦余弦嵌入的松鼠搜索算法(SCSSA)[16]. 其中，AWPSO 设计自适应加权策略，利用曲线函数映射自适应调节粒子加速系数. MPSO 设计随机和主流学习策略替代自我和全局学习策略. PECSO 融合自由分组与个体再生机制实现多样化搜索. SCSSA 将正弦余弦算法与松鼠搜索算法混合以结合二者优势.

为保证实验的公平性，所有算法的通用参数均设置为智能体个数 N=50，最大迭代次数 $t_{\max}$=1 000，测试维度 D=30. 此外，每种算法的特殊参数设置列于表 2.

表 2　各算法的特殊参数

Table 2　Special parameters of each algorithm

算法	参数
AWPSO	$a=0.000\ 035\times m$，$b=0.5$，$c=0$，$d=1.5$
MPSO	$P_i=\exp\{\mathrm{fit}(X_i(t))\}/\exp\left\{N^{-1}\sum_{i=1}^{N}\mathrm{fit}(X_i(t))\right\}$
PECSO	$\eta=0.5$，$\alpha=1$
SCSSA	$k=12$
HTPSO	$\eta=0.6$

在每个基准测试函数上，均进行30次独立重复实验，以减少随机因素对测试结果的影响. 采用30次独立实验测试结果的平均适应度值与标准差为评价指标.

表3为HTPSO算法与4种对比算法在基准函数上的测试结果. 由表3可见，HTPSO算法在6个基准函数上均获得了最小的平均适应度值和标准差，表明HTPSO算法的收敛精度和稳定性较高.

表3 不同算法的基准函数测试结果对比

Table 3 Comparison of benchmark function test results of different algorithms

函数	HTPSO		MPSO		AWPSO		SCSSA		PECSO	
	平均值	标准差	平均值	标准差	平均值	标准差	平均值	标准差	平均值	标准差
F_1	1.67×10^{3}	1.89×10^{3}	2.15×10^{8}	2.00×10^{8}	4.53×10^{7}	2.63×10^{7}	8.02×10^{3}	3.63×10^{3}	4.60×10^{3}	5.14×10^{3}
F_2	1.48×10^{4}	5.50×10^{3}	7.19×10^{4}	1.76×10^{4}	1.59×10^{4}	8.12×10^{3}	5.14×10^{4}	7.66×10^{3}	1.58×10^{4}	6.14×10^{3}
F_3	1.98×10^{5}	1.00×10^{5}	6.66×10^{6}	8.02×10^{6}	3.65×10^{7}	4.46×10^{7}	1.05×10^{7}	6.58×10^{6}	4.63×10^{6}	6.48×10^{6}
F_4	6.71×10^{3}	4.65×10^{3}	1.71×10^{4}	1.10×10^{4}	5.67×10^{5}	3.52×10^{3}	5.58×10^{4}	2.87×10^{4}	1.82×10^{4}	1.39×10^{4}
F_5	7.35×10^{3}	3.83×10^{3}	3.00×10^{5}	3.31×10^{5}	1.38×10^{4}	1.78×10^{4}	3.52×10^{5}	4.18×10^{5}	6.55×10^{4}	5.43×10^{4}
F_6	1.49×10^{5}	8.76×10^{4}	8.38×10^{5}	7.98×10^{5}	5.80×10^{5}	5.22×10^{4}	1.51×10^{6}	1.29×10^{6}	1.24×10^{6}	1.22×10^{6}

图3为6种算法在不同基准函数上的收敛曲线，为更显著地展示不同算法在多轮独立重复实验中的收敛趋势，采用平均适应度值的对数值作为性能指标. 由图3可见，HTPSO算法曲线下降速率较快且收敛位置较低，表明其能较好平衡算法勘探与开发的关系，具有较高的收敛速度和精度，进一步证明了该算法的良好优化能力.

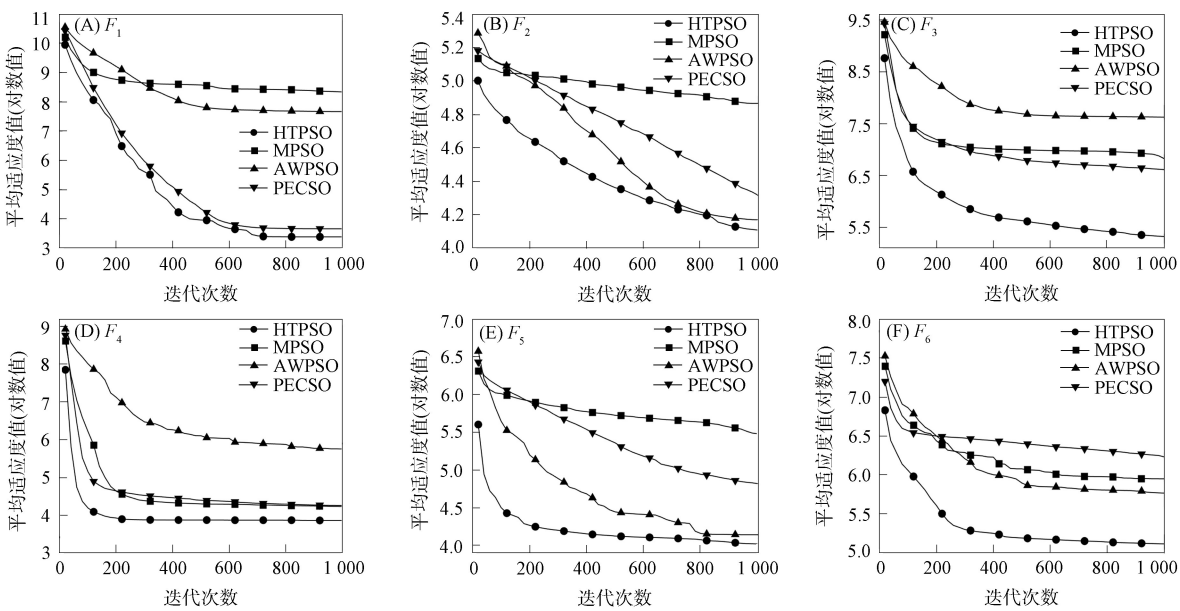


图3 6种算法在不同基准函数上的收敛曲线

Fig. 3 Convergence curves of six algorithms on different benchmark functions

3.2 工程优化问题求解

工程优化问题通过将实际工程设计问题转化为最小化优化问题进行求解，群体智能优化算法已被验证可以为其设计提供有效方案. 下面考虑3个有表示性的有约束工程优化问题：三杆桁架设计[17]、压力容器设计[18]和拉力/压缩弹簧设计[19]. 3个工程设计优化问题的基本信息列于表4.

采用HTPSO算法进行工程优化问题求解实验，对比算法实验参数设置与基准函数实验相同. 3个工程优化问题的测试结果对比列于表5，其中优化结果的最优值、最差值、平均值和标准差由重复进行的30次独立实验获得，优化变量为对应算法取得的最优值. 由表5可见，HTPSO算法在3个工程优化问题的各项性能指标中展现出较高的综合性能，优于其他3种对比算法. 表明HTPSO算法能

有效解决一些实际应用中的工程优化问题，有广泛的适用性和有效性.

表 4　各工程优化问题的基本信息

Table 4　Basic information of each engineering optimization problem

函数	问题名称	数学表达式	约束条件	可行域
F_7	三杆桁架设计	$F_8(x)=(2\sqrt{2}x_1+x_2)\times l$	$\left(\sqrt{2}x_1+\frac{x_2}{\sqrt{2}x_1^2+2x_1x_2}\right)P-\sigma\leqslant 0$ $\left(\frac{1}{\sqrt{2}x_1^2}+x_1\right)P-\sigma\leqslant 0$	$0\leqslant x_1\leqslant 1$ $l=100$ cm $P=2$ kN/cm^2 $\sigma=2$ kN/cm^2
F_8	压力容器设计	$F_9(x)=0.622\,4x_1x_3x_4+1.778\,1x_2x_3+3.166\,1x_1^2x_4+19.84x_1^2x_3$	$-x_1+0.019\,3x_3\leqslant 0$ $-x_2+0.009\,54x_3\leqslant 0$ $-\pi x_3^2x_4-4\pi x_3^3/3\leqslant -1\,960\,000$ $-x_4-240\leqslant 0$	$0\leqslant x_1,x_2\leqslant 100$ $10\leqslant x_3,x_4\leqslant 200$ x_1,x_2 是 0.062 5 的倍数
F_9	拉力/压缩弹簧	$F_{10}(x)=(x_3+2)x_2x_1^2$	$1-\frac{x_2^3x_3}{71.785x_1^4}\leqslant 0$ $4x_2^2-\frac{x_1x_2}{12.566(x_2x_1^3-x_1^4)}+\frac{1}{5.108x_1^2}\leqslant 1$ $1-\frac{x_2^3x_3}{71.785x_1^4}\leqslant 0$ $1-\frac{x_2^3x_3}{71.785x_1^4}\leqslant 0$	$0\leqslant x_1\leqslant 1$ $l=100$ cm $P=2$ kN/cm^2 $\sigma=2$ kN/cm^2

表 5　3 个工程优化问题的测试结果对比

Table 5　Comparison of test results for three engineering optimization problems

函数	算法	优化结果				优化变量			
		最优值	最差值	平均值	标准差	x_1	x_2	x_3	x_4
F_7	HTPSO	263.895 8	263.905 7	263.897 4	0.002 6	0.788 6	0.408 6	—	—
	MPSO	263.896 0	263.918 9	263.899 7	0.005 2	0.787 2	0.412 5	—	—
	AWPSO	263.895 8	263.895 9	263.895 9	0.000 1	0.788 7	0.408 2	—	—
	PECSO	263.895 9	265.775 6	264.198 6	0.460 4	0.788 5	0.408 8	—	—
F_8	HTPSO	6 059.714 3	7 273.511 1	6 328.204 0	346.426 4	13.150 2	7.117 0	42.098 4	176.636 6
	MPSO	6 370.779 8	7 544.492 5	6 740.725 6	326.029 9	15.918 4	8.027 6	51.813 5	84.578 6
	AWPSO	6 059.714 3	7 273.511 1	6 360.811 6	374.837 4	12.507 7	7.211 8	42.098 4	176.636 6
	PECSO	6 059.714 3	7 337.490 4	6 355.173 8	376.004 5	12.807 4	6.767 2	42.098 4	176.636 6
F_9	HTPSO	0.012 7	0.014 7	0.013 2	0.000 6	0.053 2	0.395 0	9.354 0	—
	MPSO	0.012 8	0.025 2	0.014 9	0.002 7	0.050 0	0.317 1	14.212 0	—
	AWPSO	0.012 8	0.015 4	0.014 0	0.013 3	0.051 6	0.354 3	11.490 9	—
	PECSO	0.012 7	0.016 3	0.013 2	0.000 8	0.051 8	0.359 0	11.155 7	—

综上所述，针对粒子群优化算法在解决复杂工程优化问题时存在搜索效率低和易陷入局部最优的问题，本文提出了一种基于屋型拓扑的粒子群优化(HTPSO)算法. 通过提出屋型拓扑结构和适应于该拓扑结构的改进位置更新策略，拓展了粒子群中粒子的知识来源，增强了算法位置更新的泛化性，平衡了种群勘探能力和开发能力，缓解了 PSO 算法在全局探索能力、搜索效率和收敛精度方面的不足. 在 3 个工程优化问题的求解中，仿真结果表明，HTPSO 算法相比于其他算法表现出求解精度高和稳定性强的优势，证明了其在解决实际优化问题上的可行性.

参　考　文　献

[1] PAN J S, HU P, SNÁŠEL V, et al. A Survey on Binary Metaheuristic Algorithms and Their Engineering Applications [J]. Artificial Intelligence Review, 2023, 56(7): 6101-6167.

[2] QU L T, HE W B, LI J F, et al. Explicit and Size-Adaptive PSO-Based Feature Selection for Classification [J].

Swarm and Evolutionary Computation, 2023, 77: 101249-1-101249-13.

[3] 管仁初, 贺宝润, 梁艳春, 等. 基于亲缘关系选择的粒子群优化算法[J]. 吉林大学学报(工学版), 2022, 52(8): 1842-1849. (GUAN R C, HE B R, LIANG Y C, et al. Particle Swarm Optimization Algorithm Based on Kinship Selection [J]. Journal of Jilin University (Engineering and Technology Edition), 2022, 52(8): 1842-1849.)

[4] LI T Y, SHI J Y, DENG W, et al. Pyramid Particle Swarm Optimization with Novel Strategies of Competition and Cooperation [J]. Applied Soft Computing, 2022, 121: 108731-1-108731-22.

[5] WANG Z W, ZHANG W T, GUO Y N, et al. A Multi-objective Chicken Swarm Optimization Algorithm Based on Dual External Archive with Various Elites [J]. Applied Soft Computing, 2023, 133: 109920-1-109920-24.

[6] RIZK-ALLAH R, HAGAG E, EL-FERGANY A. Chaos-Enhanced Multi-objective Tunicate Swarm Algorithm for Economic-Emission Load Dispatch Problem [J]. Soft Computing, 2023, 27(9): 5721-5739.

[7] 兰森淼, 胡黄水, 王婷婷, 等. 基于改进鲸鱼优化PID的无刷直流电机转速控制算法[J]. 吉林大学学报(理学版), 2024, 62(3): 704-712. (LAN M M, HU H S, WANG T T, et al. Speed Control Algorithm of Brushless DC Motor Based on Improved Whale Optimization PID [J]. Journal of Jilin University (Science Edition), 2024, 62(3): 704-712.)

[8] HONG L B, YU X M, WANG B, et al. An Improved Ensemble Particle Swarm Optimizer Using Niching Behavior and Covariance Matrix Adapted Retreat Phase [J]. Swarm and Evolutionary Computation, 2023, 78: 101278-1-101278-20.

[9] 李俊祺, 林伟伟, 石方, 等. 基于混合群智能的节能虚拟机整合方法[J]. 软件学报, 2022, 33(11): 3944-3966. (LI J Q, LIN W W, SHI F, et al. Energy Efficient Hybrid Swarm Intelligence Virtual Machine Consolidation Method [J]. Journal of Software, 2022, 33(11): 3944-3966.)

[10] TANG C J, SUN W, XUE M, et al. A Hybrid Whale Optimization Algorithm with Artificial Bee Colony [J]. Soft Computing, 2022, 26(5): 2075-2097.

[11] PENG H, ZENG Z G, DENG C S, et al. Multi-strategy Serial Cuckoo Search Algorithm for Global Optimization [J]. Knowledge-Based Systems, 2021, 214: 106729-1-106729-19.

[12] WU G H, MALLIPEDDI R, SUGANTHAN P. Problem Definitions and Evaluation Criteria for the CEC 2017 Competition on Constrained Real-Parameter Optimization [R]. Changsha: National University of Defense Technology, 2017.

[13] LIU W B, WANG Z D, YUAN Y, et al. A Novel Sigmoid-Function-Based Adaptive Weighted Particle Swarm Optimizer [J]. IEEE Transactions on Cybernetics, 2019, 51(2): 1085-1093.

[14] LIU H, ZHANG X W, TU L P. A Modified Particle Swarm Optimization Using Adaptive Strategy [J]. Expert Systems with Applications, 2020, 152: 113353-1-113353-19.

[15] ZHANG Y F, WANG L M, ZHAO J P. PECSO: An Improved Chicken Swarm Optimization Algorithm with Performance-Enhanced Strategy and Its Application [J]. Biomimetics, 2023, 8(4): 355-1-355-24.

[16] ZENG L, SHI J Y, LI M, et al. Sine Cosine Embedded Squirrel Search Algorithm for Global Optimization and Engineering Design [J]. Cluster Computing, 2023, 27: 4415-4448.

[17] WANG Z W, QIN C, WAN B T, et al. An Adaptive Fuzzy Chicken Swarm Optimization Algorithm [J]. Mathematical Problems in Engineering, 2021, 2021: 8896794-1-8896794-17.

[18] MOOSAVI S, BARDSIRI V. Poor and Rich Optimization Algorithm: A New Human-Based and Multi Populations Algorithm [J]. Engineering Applications of Artificial Intelligence, 2019, 86: 165-181.

[19] XU X L, HU Z B, SU Q H, et al. Multivariable Grey Prediction Evolution Algorithm: A New Metaheuristic [J]. Applied Soft Computing, 2020, 89: 106086-1-106086-15.

(责任编辑: 韩 啸)